

COMUNICACIÓN CON MARCADORES ALFANUMÉRICOS MP ELECTRONICS

Manual Técnico

Revisión del Documento 2.0

Fecha: 05/07/2013
Nombre de fichero: Comunicación con Marcadores Alfanuméricos MP Electronics. Rev. 2.0

ÍNDICE GENERAL

1. DESCRIPCIÓN

Con los Marcadores Numéricos de MP Electronics, se puede interactuar mediante varios Protocolos de Comunicación. En este documento, se detalla la comunicación para los Protocolos ELEKTRA, MODBUS y ASCII.

Para que el Marcador responda a un determinado protocolo, salvo el Protocolo ELEKTRA, éste debe estar habilitado en la configuración del dispositivo. El Protocolo ELEKTRA (propio de MP Electronics) se encuentra habilitado por defecto, y en caso de desearlo, podemos deshabilitarlo. Incluso en caso de deshabilitarse, éste permanecerá operativo a través del puerto USB del Marcador, dado que es el protocolo con que podemos modificar la configuración del dispositivo.

El Protocolo ELEKTRA, que se debe utilizar en combinación con el Lenguaje Script Homónimo, es el Protocolo Nativo de los Marcadores Numéricos de MP Electronics. Dicho Protocolo dispone de multitud de Órdenes y opciones, dado que su uso se extiende más allá de los Marcadores Numéricos. En éste documento solo nos referiremos a las Órdenes y opciones necesarias para interactuar con Marcadores Numéricos.

El protocolo MODBUS (tanto en modo RTU como en modo TCP/IP). Este protocolo es muy utilizado en el entorno industrial y fácilmente adaptable a muchos tipos de instrumentación, como por ejemplo los Controladores Lógicos Programables (PLC).

Este protocolo permite comunicar fácilmente con cualquier equipo que disponga de una línea serie y en el que se pueda configurar el protocolo, como puede un PLC o un PC con un software que permita el envío/recepción de Bytes por un puerto serie.

2. PROTOCOLO Y SCRIPT ELEKTRA

2.1. Protocolo ELEKTRA.

2.1.1. Descripción del Protocolo.

El Protocolo de Paquetes ELEKTRA (en adelante ***PPE***) es el Protocolo Nativo de los Marcadores Alfanuméricos y de otros dispositivos de MP Electronics. Está destinado a la comunicación entre diferentes dispositivos, un equipo ***Maestro*** y/o Host, y otro(s) dispositivo(s) ***Esclavo*** y/o Device, para su programación, configuración y actualización. El ámbito de este Protocolo abarca los sistemas más usuales de comunicación entre sistemas electrónicos, como son (sin exclusión de otros futuros): Serie RS232 y RS485, USB1.0 y 2.0, Ethernet, Wifi.

El protocolo está basado en un sistema Unicast (punto a punto) half-duplex, con una configuración Maestro-Esclavo, es decir, en ningún caso el dispositivo Esclavo podrá establecer una comunicación sin que el Master le "hable" primero. Los Marcadores Alfanuméricos de MP Electronics son Esclavos para el PPE.

En casos particulares también se puede trabajar con un sistema Multicast.

La conectividad entre el Marcador (Esclavo) y el dispositivo que realice la función de Maestro, puede ser por varios puertos de comunicación, tal y como se ilustra en la siguiente tabla.

Puerto	*Protocolo de Paquetes ELEKTRA (PPE)*	
	Conectividad	*Habilitado por defecto*
RS232	SI	SI
RS485	SI	SI
USB	USB Device – Función CDC (VCP)	SI (No se puede deshabilitar)
ETHERNET (TCP)	Hay que abrir el Socket del Puerto 53 con la IP del Dispositivo	SI

Tabla 1: Opciones de conectividad para el PPE

2.1.2. Estructura del Protocolo.

Cada paquete de del PPE está formado por 3 bloques fundamentales.

- “Header” o Cabecera
- Datos (opcionalmente)
- “Checksum”

En la siguiente figura se puede apreciar de una forma gráfica su composición.



Figura 1: Composición del PPE

2.1.3. “Header” del Protocolo.

El Header del Protocolo (a partir de ahora, denominaremos 'Header') está formado por una trama de 20 bytes. Dentro de estos bytes nos encontramos con una serie de Token (componente léxico que tiene un significado coherente en un cierto lenguaje de programación) para sincronizar, verificar la trama, la dirección de origen, la dirección de destino y su máscara. Es requerimiento básico que cada dispositivo de la red tenga su dirección[1] única.

Cada dispositivo solo puede responder a un paquete si este está dirigido a su dirección, mientras que cada dispositivo solo puede ejecutar la orden contenida al paquete si, y solo si, la dirección

[1] Las direcciones de los dispositivos se componen de 2 bytes. El MSB es el Numero de Grupo (1 a 223) y el LSB es el Numero de Dispositivo (1 a 253).

de destino es igual a la dirección del dispositivo una vez aplicada la máscara de la dirección destino. En el caso de la respuesta, cuando el dispositivo se comunica con el host, normalmente un PC, este debe verificar también que el paquete va dirigido a él.

En las siguientes tablas se detallan los campos del Header del PPE.

	# BYTE	VALOR	NOMBRE	DESCRIPCIÓN
HEADER	**1**	*0xAA*	**TOKEN 1**	**TOKEN 1**
	2	*0x55*	**TOKEN 2**	**TOKEN 2**
	3	ADO1	**AddrOrg**	MSB dirección dispositivo origen
	4	ADO2	**AddrOrg**	LSB dirección dispositivo origen
	5	ADE1	**AddrDest**	MSB dirección dispositivo destino
	6	ADE2	**AddrDest**	LSB dirección dispositivo destino
	7	ADM1	**AddrMask**	MSB mascara dirección
	8	ADM2	**AddrMask**	LSB mascara dirección
	9	OD1	**OD1**	MSB orden
	10	OD2	**OD2**	LSB orden
	11	PK1	**PacketNum**	MSB numero paquete
	12	PK2	**PacketNum**	LSB numero paquete
	13	NB1	**PacketSize**	MSB numero bytes paquete datos
	14	NB2	**PacketSize**	LSB numero bytes paquete datos
	15	*0xBB*	**TOKEN 3**	**TOKEN 3**
	16	*0x99*	**TOKEN 4**	**TOKEN 4**
	17	*0x77*	**TOKEN 5**	**TOKEN 5**
	18	*0xFF*	**TOKEN 6**	**TOKEN 6**
	19	*0x00*	**TOKEN 7**	**TOKEN 7**
	20	*0xFF*	**TOKEN 8**	**TOKEN 8**
DATOS	**21**	DATO[0]	**Primer byte de datos**	Primer byte de datos (si hay datos)
	…	…	**…**	…
	20+n [2]	DATO[n-1]	**Ultimo byte de datos**	Ultimo byte de datos (si hay datos)
CHECKSUM	**20+n+1**	MSB Checksum	**MSB del Checksum**	
	20+n+2	LSB Checksum	**LSB del Checksum**	

Tabla 2: Estructura del Protocolo EPP

[2] Donde n = número de bytes de datos. En caso de n=0, el MSB del Checksum quedará en el #BYTE 21 y el LSB del Checksum quedará en el #BYTE 22

- **AddrOrg:** Dirección de Origen del Mensaje. Los Marcadores Alfanuméricos solo responden a mensajes de la dirección ***0xF000*** (Grupo 240, Dispositivo 0).
- **AddrDest:** Dirección de Destino del Mensaje.
- **AddrMask:** Éste campo, se usa para las Ordenes enviadas en modo Multicast. En caso de trabajar en modo Unicast (habitual) debe tener como valor ***0xFFFF***.
- **OD1:** Orden (1 Byte).
- **OD2:** Sub-Orden (1 Byte).
- **PacketNum:** Número de Paquete.
- **PacketSize:** Numero de bytes del campo *Datos*. En caso de no haber datos, éste campo valdrá 0.

Se debe verificar la dirección de origen **AddrOrg**, la de destino **AddrDest** y el **PacketNum**.

Para cada paquete con una orden, salvo que se trabaje en modo multicast, el dispositivo responde con un paquete ACK o NACK, según haya podido ejecutar correctamente la orden o no.

Toda respuesta de cada orden (ACK/NACK) debe tener el mismo **PacketNum** que el paquete de la orden; en caso contrario la respuesta se considerará no valida. El **PacketNum** se incrementará por nuevo paquete, incluso por cada reintento, es decir, no habrá 2 órdenes con el mismo **PacketNum** de forma consecutiva.

El Header[3] de Respuesta tiene los mismos campos que el de la orden, con una salvedad. El campo OD2 es substituido por el campo ACK/NACK. Para paquetes ACK el valor es ***0x06***, mientras que para NACK es ***0x0F***.

2.1.4. Campo de Datos.

El campo de datos puede contener diferente información según la Orden de que se trate.

En las respuestas ACK[4], el campo **PacketSize** nos indica el número de bytes de datos que se envían a continuación del Header (sin espera adicional).

En las respuestas NACK, a continuación del Header se envían una serie de datos que nos indican el tipo de error que ha causado el NACK:

PacketSize = 0x0010, es decir, nos envía 16 (decimal) bytes:

Dato[0]: LSB Tipo error
Dato[1]: MSB Tipo error
Dato[2]: Sub-Orden que ha generado el error

[3] Tener en cuenta que para el paquete de respuesta, ***AddrOrg*** es la dirección del dispositivo que responde (Esclavo), mientras que ***AddrDest*** es la dirección del Maestro que envió la orden.

[4] Una parte importante de las respuestas con ACK, no contienen datos adicionales, o sea, el ***PacketSize*** es 0

Dato[3]: Orden que ha generado el error
Dato[4] a Dato[15] : no asignado

2.1.5. Checksum.

Al final del paquete, se tiene que enviar el **Checksum** para verificar que no ha habido errores durante la transmisión. Si este **Checksum** no corresponde, el destinatario descartara el paquete, y el origen, al terminar el tiempo de espera para el ACK, volverá a enviar el mismo paquete con el **PacketNum** incrementado. Si el paquete contiene datos, estos están posicionados después del Header y antes del **Checksum**. La fórmula para el cálculo del **Checksum** es la siguiente:

$$\text{Checksum} = (\text{WORD})(\sim(\sum(\text{Header} + \text{Datos}))) + 1$$

2.1.6. Órdenes relevantes para la comunicación con Marcadores Alfanuméricos.

2.1.6.1. Envío de Script al Marcador.

ÓRDEN: PROGRAMA DE SCRIPT (***0xB0***)

SUB-ORDEN *0x50*: Ejecución inmediata del Script contenido (en datos del paquete de la orden).

Datos:

Nombre	**Descripción**	**Offset**	**Medida (Bytes)**
SCRIPT_HEADER	Cabecera del Script	0	150
SCRIPT	Script	150	Variable

SCRIPT_HEADER → 150 Bytes: Rellenar con 0s (0x00)

SCRIPT[5] → variable (ver descripción del script). Terminado en carácter NULL (0x00) para detectar el final del script.

Respuesta: ACK (sin datos) / NACK.

2.1.6.2. Detención de los Scripts del Marcador (Borrado de todas las Líneas).

ÓRDEN: PROGRAMA DE SCRIPT (***0xB0***)

[5] El Script debe empezar obligatoriamente por '~', i el último byte debe ser 0 (final de script). Ver capitulo siguiente.

SUB-ORDEN *0x70*: Detiene la ejecución de todos los Scripts del Marcador.

Datos: *Ninguno*

Respuesta: ACK (sin datos) / NACK.

2.2. Script ELEKTRA.

Para poder representar en un marcador un dato cualquiera con un efecto deseado, tendremos que comunicarnos con el Marcador y usar un lenguaje característico para su debida interpretación. Ese lenguaje característico es el “Script”, que en este caso recibe el nombre de Script Elektra.

La estructura básica del Script es ~ ORDENES ~ TEXTO, y finalizado con \0.

El TEXTO no es imprescindible, pero sí las ORDENES

Para la programación de los Marcadores Alfanuméricos, debemos tener en cuenta las especificaciones detalladas a continuación.

2.2.1. Símbolos y Caracteres del Script Elektra para programación de los Marcadores.

- **/0 →** Significa el carácter NULL, 0 decimal o 0x00 hexadecimal, en finales de cadena.
- **~ →** Carácter 126 decimal o 0x7E hexadecimal.
- ‘0’ a ‘9’ → Caracteres ASCII '0' a '9' (código hexadecimal 0x30 a 0x39)
- ‘A’ a ‘Z’ → Caracteres ASCII 'A' a 'Z' (código hexadecimal 0x41 a 0x5A)
- ‘a’ a ‘z’ → Caracteres ASCII 'a' a 'z' (código hexadecimal 0x61 a 0x7A)

La marca de final de Script es el carácter NULL (valor=0), similar al de una cadena de caracteres en lenguaje C.

2.2.2. Órdenes del Script Elektra usadas para programación de los Marcadores.

Este Script dispone de una multitud de órdenes que soportan múltiples opciones, efectos de texto y funciones que los Marcadores Alfanuméricos no necesitan. Por tal motivo procedemos a detallar cuales son las órdenes que debemos usar para representar valores en nuestros marcadores. Debemos tener en cuenta también que el Script debe siempre terminar con el carácter NULL

DESCRIPCIÓN	VALOR	HEX.
Establece nº de Línea.	‘L’	0x4C
Establece Luminosidad	'A'	0x41
Activa/Desactiva Intermitente	‘B’	0x42
Establece Color texto	‘R’	0x52

Tabla 3: Tokens del Script Elektra usados en Marcadores Alfanuméricos

- **Establece nº de Línea.** Esta Orden indica la línea de donde se efectuará la visualización de los datos alfanuméricos.

  Parámetros: Número de línea, de '1' hasta el límite del Marcador, en ASCII. Las líneas del Marcador se numeran de 1 al número total de líneas, siendo la línea 1 la situada en la parte superior del marcador.

  Ejemplo: Línea 1 → ~L1~

- **Establece Luminosidad.** Esta Orden incluida en el Script, establece la intensidad lumínica (en % respecto a la intensidad máxima) con que todo el marcador representa los datos. Su inclusión está destinada a poder variar la intensidad en un momento dado. La inclusión de esta orden, anula la configuración de luminosidad del control del marcador mientras se ejecuta, excepto en los límites de Luminosidad Máxima y mínima de la dicha configuración. El valor '0' indica Luminosidad Automática en función de la Luz Ambiente. (solo en aquellos productos con sonda de luz incorporada).

  Parámetros: Valor de '10' a '100' (en ASCII)

  Ejemplo: Establecer Luminosidad al 25 % → ~A25~

  Ejemplo: Establecer Luminosidad Automática en función de Luz Ambiente → ~A0~

- **Activa/Desactiva Intermitente.** Esta Orden determina el efecto intermitente del texto, su funcionamiento es tipo báscula o 'toggle', o sea, cada vez que se encuentra, cambia. Por ejemplo, la primera vez, activa el texto que sigue en intermitencia, y la segunda lo vuelve a dejar en fijo. Su estado se reinicia al establecer nueva línea, es decir, al definir una nueva línea o ventana, empieza siempre desactivado.

  Parámetros:

  - Ninguno para Intermitencia Normal.
  - Carácter 'f' para Intermitencia Rápida.

- **Establece Color Texto.** Esta Orden determina el color en que se visualizarán el texto, en Marcadores con más de 1 color.

  Parámetros: Variable (en ASCII), de '0' a '3' o número máximo de colores. El '0' corresponde a un color y es apagado. Valor por defecto es '1'

2.2.3. Ejemplos de Scripts.

Para entender mejor el Script vamos a poner algunos casos como ejemplo.

- Ejemplo 1: Si queremos representar el valor numérico 1.387 en la línea 1 (la superior) del marcador, el *Script* sería:

  En ASCII: ~L1~1.387**\0**

  En Hexadecimal: 0x7E 0x4C 0x31 0x7E 0x31 0x2E 0x33 0x38 0x37 0x00

- Ejemplo 2: Si queremos representar el valor numérico 1.387 en la línea 1 y el texto "Hola" en la línea 2 del marcador, el *Script* sería:

En ASCII: ~L1~1.387~L2~Hola**\0**

En Hexadecimal:

0x7E 0x4C 0x31 0x7E 0x31 0x2E 0x33 0x38 0x37 0x7E 0x4C 0x32 0x7E 0x48 0x6F 0x6C 0x61 0x00

- Ejemplo 3: Si queremos representar el valor numérico 1.387 en la línea 1 y el texto “Hola” en la línea 2 del marcador, y además fijar la luminosidad del marcador al 50%, el *Script* sería:

En ASCII: ~L1~1.387~L2~ Hola ~A50~**\0**

- Ejemplo 4: Si queremos representar el valor numérico 1.387 en la línea 1 y el texto “Hola” en la línea 2 del marcador, fijar la luminosidad del marcador al 50%, y que la línea 1 este en modo intermitente, el *Script* sería:

En ASCII: ~L1~~B~1.387~B~~L2~Hola~A50~**\0**

- Ejemplo 5: Si queremos representar el valor numérico 1.387 en la línea 1 y el texto “Hola” en la línea 2 del marcador, fijar la luminosidad del marcador al 50%, y que en la línea 1 este en intermitente solamente los dígitos “387”, el *Script* sería:

En ASCII: ~L1~1. ~B~387~B~~L2~Hola~A50~**\0**

- Ejemplo 6: Si queremos representar el valor numérico 1.387 en la línea 1 y el texto “Hola” en la línea 2 del marcador, fijar la luminosidad del marcador al 50%, y que la línea 1 este en modo intermitente rápido, el *Script* sería:

En ASCII: ~L1~~Bf~1.387~B~~L2~Hola~A50~**\0**

2.3. Ejemplos.

Para los siguientes casos, se muestra como sería la Trama del Protocolo ELEKTRA enviada:

- Ejemplo 1: Para que se visualice en la línea 1 (la superior en caso de haber más de una) del Marcador con dirección 1.1 (Grupo 1, Terminal 1), el texto "Hola", se deberá enviar la siguiente Trama.

	BYTE	*VALOR*		DESCRIPCIÓN
HEADER	1	*0xAA*	**TOKEN 1**	**TOKEN 1**
	2	*0x55*	**TOKEN 2**	**TOKEN 2**
	3	*0xF0*	**AddrOrg**	Dirección de origen del Host: 0xF000 (240.0)
	4	*0x00*	**AddrOrg**	
	5	*0x01*	**AddrDest**	Dirección del Marcador: 0x0101 (1.1)
	6	*0x01*	**AddrDest**	
	7	*0xFF*	**AddrMask**	Mascara de Dirección: 0xFFFF si en modo Unicast
	8	*0xFF*	**AddrMask**	
	9	*0xB0*	**OD1: Orden**	Orden: PROGRAMA DE SCRIPT
	10	*0x50*	**OD2: Sub-Orden**	Sub-Orden: Ejecución inmediata del Script contenido
	11	*0x00*	**PacketNum**	Número del Paquete.
	12	*0x01*	**PacketNum**	
	13	*0x00*	**PacketSize**	Número de bytes de datos: 0x009F (159 Bytes) (150+9)
	14	*0x9F*	**PacketSize**	
	15	*0xBB*	**TOKEN 3**	**TOKEN 3**
	16	*0x99*	**TOKEN 4**	**TOKEN 4**
	17	*0x77*	**TOKEN 5**	**TOKEN 5**
	18	*0xFF*	**TOKEN 6**	**TOKEN 6**
	19	*0x00*	**TOKEN 7**	**TOKEN 7**
	20	*0xFF*	**TOKEN 8**	**TOKEN 8**
DATOS	21	*0x00*	**Primer byte de datos**	SCRIPT_HEADER (150 Bytes)
	…	*0x00*	**…**	
	170	*0x00*	**…**	
	171	*0x7E*		'~'
	172	*0x4C*		'L'
	173	*0x31*		'1'
	174	*0x7E*		'~'
	175	*0x48*		'H'
	176	*0x6F*		'o'
	177	*0x6C*		'l'
	178	*0x61*		'a'
	179	*0x00*	**Último byte de datos**	NULL
CHECKSUM	180	*0xF3*	**MSB del Checksum**	*MSB Checksum*
	181	*0xAB*	**LSB del Checksum**	*LSB Checksum*

Ejemplo 1: Trama Enviada al Marcador para visualizar "Hola" en la línea 1

Y el Marcador deberá responder:

	BYTE	VALOR		DESCRIPCIÓN
HEADER	1	0xAA	TOKEN 1	TOKEN 1
	2	0x55	TOKEN 2	TOKEN 2
	3	0x01	AddrOrg	Dirección del Marcador: 0x0101 (1.1)
	4	0x01	AddrOrg	
	5	0xF0	AddrDest	Dirección de origen del Host: 0xF000 (240.0)
	6	0x00	AddrDest	
	7	0xFF	AddrMask	Mascara de Dirección: 0xFFFF si en modo Unicast
	8	0xFF	AddrMask	
	9	0xB0	OD1: Orden	Orden: PROGRAMA DE SCRIPT
	10	0x06	OD2: ACK/NACK	ACK
	11	0x00	PacketNum	Número del Paquete.
	12	0x01	PacketNum	
	13	0x00	PacketSize	Número de bytes de datos: 0 (sin datos)
	14	0x00	PacketSize	
	15	0xBB	TOKEN 3	TOKEN 3
	16	0x99	TOKEN 4	TOKEN 4
	17	0x77	TOKEN 5	TOKEN 5
	18	0xFF	TOKEN 6	TOKEN 6
	19	0x00	TOKEN 7	TOKEN 7
	20	0xFF	TOKEN 8	TOKEN 8
CHECKSUM	21	0xF7	MSB del Checksum	MSB Checksum
	22	0x91	LSB del Checksum	LSB Checksum

Ejemplo 1: Respuesta del Marcador

- Ejemplo 2: Para que se borren todos los datos del Marcador con dirección 1.1 (Grupo 1, Terminal 1), se deberá enviar la siguiente Trama.

	BYTE	VALOR		DESCRIPCIÓN
HEADER	1	0xAA	TOKEN 1	TOKEN 1
	2	0x55	TOKEN 2	TOKEN 2
	3	0xF0	AddrOrg	Dirección de origen del Host: 0xF000 (240.0)
	4	0x00	AddrOrg	
	5	0x01	AddrDest	Dirección del Marcador: 0x0101 (1.1)
	6	0x01	AddrDest	
	7	0xFF	AddrMask	Mascara de Dirección: 0xFFFF si en modo Unicast
	8	0xFF	AddrMask	
	9	0xB0	OD1: Orden	Orden: PROGRAMA DE SCRIPT
	10	0x70	OD2: Sub-Orden	Sub-Orden: Detener la ejecución de todos los Scripts
	11	0x00	PacketNum	Número del Paquete.
	12	0x02	PacketNum	
	13	0x00	PacketSize	Número de bytes de datos: 0 (sin datos)
	14	0x00	PacketSize	
	15	0xBB	TOKEN 3	TOKEN 3
	16	0x99	TOKEN 4	TOKEN 4

	17	*0x77*	**TOKEN 5**	**TOKEN 5**
	18	*0xFF*	**TOKEN 6**	**TOKEN 6**
	19	*0x00*	**TOKEN 7**	**TOKEN 7**
	20	*0xFF*	**TOKEN 8**	**TOKEN 8**
CHECKSUM	21	*0xF7*	**MSB del Checksum**	*MSB Checksum*
	22	*0x26*	**LSB del Checksum**	*LSB Checksum*

Ejemplo 2: Trama Enviada al Marcador para borrar todas las líneas

Y el Marcador deberá responder:

	BYTE	*VALOR*		DESCRIPCIÓN
HEADER	1	*0xAA*	**TOKEN 1**	**TOKEN 1**
	2	*0x55*	**TOKEN 2**	**TOKEN 2**
	3	*0x01*	**AddrOrg**	Dirección del Marcador: 0x0101 (1.1)
	4	*0x01*	**AddrOrg**	
	5	*0xF0*	**AddrDest**	Dirección de origen del Host: 0xF000 (240.0)
	6	*0x00*	**AddrDest**	
	7	*0xFF*	**AddrMask**	Mascara de Dirección: 0xFFFF si en modo Unicast
	8	*0xFF*	**AddrMask**	
	9	*0xB0*	**OD1: Orden**	Orden: PROGRAMA DE SCRIPT
	10	*0x06*	**OD2: ACK/NACK**	ACK
	11	*0x00*	**PacketNum**	Número del Paquete.
	12	*0x02*	**PacketNum**	
	13	*0x00*	**PacketSize**	Número de bytes de datos: 0 (sin datos)
	14	*0x00*	**PacketSize**	
	15	*0xBB*	**TOKEN 3**	**TOKEN 3**
	16	*0x99*	**TOKEN 4**	**TOKEN 4**
	17	*0x77*	**TOKEN 5**	**TOKEN 5**
	18	*0xFF*	**TOKEN 6**	**TOKEN 6**
	19	*0x00*	**TOKEN 7**	**TOKEN 7**
	20	*0xFF*	**TOKEN 8**	**TOKEN 8**
CHECKSUM	21	*0xF7*	**MSB del Checksum**	*MSB Checksum*
	22	*0x90*	**LSB del Checksum**	*LSB Checksum*

Ejemplo 2: Respuesta del Marcador

3. PROTOCOLO MODBUS

Con los Marcadores Numéricos de MP Electronics, se puede interactuar mediante el protocolo MODBUS (tanto en modo RTU como en modo TCP/IP). Este protocolo es muy utilizado en el entorno industrial y fácilmente adaptable a muchos tipos de instrumentación, como por ejemplo los Controladores Lógicos Programables (PLC).

Para poder comunicar mediante MODBUS con un Marcador de MP Electronics, es necesario que éste disponga de un puerto de comunicación compatible con dicho protocolo y que el Protocolo se encuentre habilitado. Para habilitar / deshabilitar el Protocolo MODBUS (tanto RTU como TCP/IP) se debe acceder a la Configuración Interna del Marcador (Parámetro Núm. 230) y activar/desactivar el *Flag* correspondiente.

La siguiente tabla muestra las variantes del protocolo MODBUS soportadas por los Marcadores, en caso de disponer éstos de los puertos de comunicación correspondientes.

Puerto	***Protocolo ModBus-RTU***		***Protocolo ModBus-TCP/IP***	
	Conectividad	***Habilitado por defecto***	***Conectividad***	***Habilitado por defecto***
RS232	SI	NO	NO	---
RS485	SI	NO	NO	---
USB	NO	---	NO	---
ETHERNET (TCP)	NO	---	SI (Puerto 502)	NO

Tabla 4: Protocolo MODBUS según puerto de comunicación del Marcador

3.1. Protocolo MODBUS-RTU.

El protocolo MODBUS en Modo RTU utiliza los silencios en la línea de transmisión para indicar los inicios y finales del mensaje. Se considera un silencio el tiempo igual o mayor al necesario para transmitir 3,5 bytes. Para cada velocidad de transmisión le corresponde un tiempo de silencio específico. Una vez realizada la transmisión de un mensaje, no se puede iniciar la transmisión de otro hasta que no haya transcurrido el tiempo de silencio necesario (3,5 veces el tiempo de transmisión de un byte).

Con este protocolo, el Marcador trabaja en modo Esclavo. Una vez recibido un mensaje con la dirección del mismo, se devolverá un mensaje con el resultado de la transmisión.

3.2. Protocolo MODBUS-TCP/IP.

El protocolo MODBUS en Modo TCP/IP (en adelante MODBUS-TCP) es una variante del Protocolo MODBUS para comunicaciones sobre redes TCP/IP, realizando las conexiones a través del puerto TCP 502.

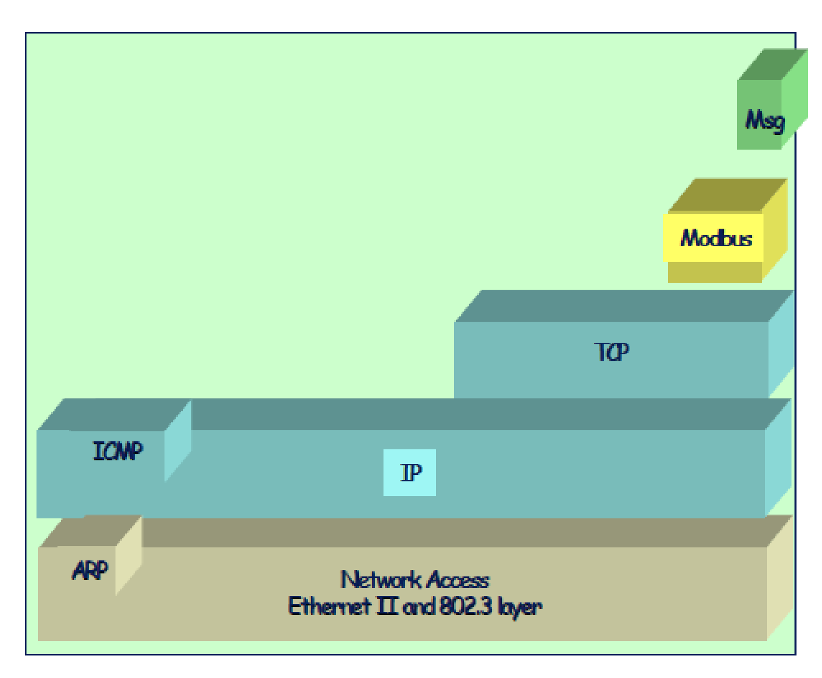
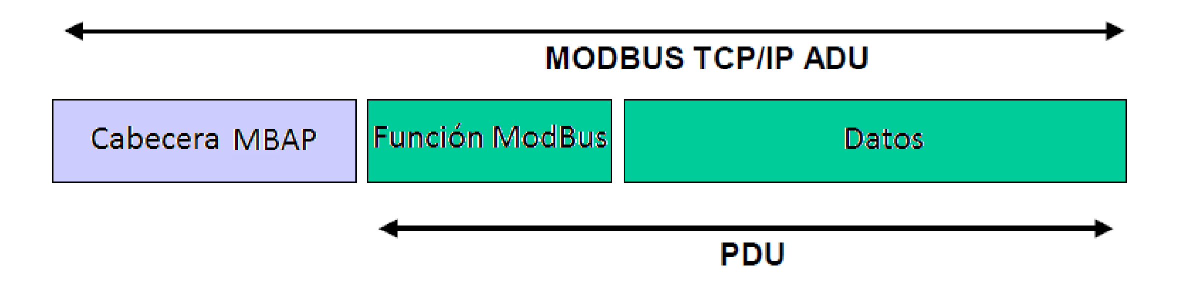


Figura 2: Esquema de niveles de comunicación para comunicaciones ModBus-TCP

Con este protocolo, el Marcador trabaja en modo Esclavo para MODBUS (Servidor para TCP). Las tramas son igual a las del protocolo MODBUS en modo RTU con las siguientes diferencias:

- El campo ID (Dirección del Dispositivo) de la trama MODBUS-RTU es substituido por una cabecera llamada Cabecera MBAP, los campos de la cual se detallan en la Tabla X.
- No hay CRC no ningún código de control de errores, dado que los protocolos de capas inferiores se ocupan de tal tarea.

La estructura de la trama es la siguiente:



La cabecera MBAP (7 Bytes) contiene los siguientes campos:

Campo	*Bytes*	*Descripción*	*Cliente (Master)*	*Servidor (Slave)*
Transaction Identifier	2	Número de la transacción	Inicializado por Cliente	Reenviado por Servidor
Protocol Identifier	2	0 = Protocolo MODBUS	Inicializado por Cliente	Reenviado por Servidor
Length	2	Numero de Bytes de la trama que siguen a éste (de "Unit Identifier" al final)	Inicializado por Cliente	Inicializado por Servidor
Unit Identifier	1	Siempre a 255 o Unit ID Pantalla	Inicializado por Cliente	Reenviado por Servidor

Tabla 5: Cabecera MBAP del Protocolo ModBus-TCP

Una vez recibido un mensaje con la dirección de la misma, se devolverá un mensaje con el resultado de la transmisión.

3.3. Funciones soportadas en comunicación por MODBUS.

Los Marcadores Numéricos de MP Electronics soportan la **Función 16 (10h) "*Write Multiple Registers*"**, de MODBUS.

A través del Protocolo MODBUS se pueden realizar las funciones que se detallan a continuación. A cada dirección de registro, le corresponde una función distinta.

3.3.1. Configuración del Marcador

Mediante MODBUS podemos modificar los siguientes parámetros de la configuración del Marcador. Dichos parámetros afectan al conjunto del dispositivo.

Dirección		**Función**	**Descripción**	***Mínimo***	***Máximo***
Dec.	*Hex.*				
1024	400	Modo de Funcionamiento	Si se escribe el valor **99** en el registro, se borra el contenido de todas las líneas del Marcador.	0	99
1025	401	Luminosidad	Establece la luminosidad del Marcador (entre 1% y 100%). El valor 0 indica Luminosidad Automática en función de la Luz Ambiente. (Solo en aquellos productos con sonda de luz incorporada).	0	100

Tabla 6: Funciones de los registros ModBus en Marcadores Alfanuméricos MP Electronics

Por ejemplo, si queremos fijar la luminosidad de nuestro Marcador al 50% de la luminosidad máxima, la PDU de MODBUS enviada sería:

Func.	Dirección Registro		Núm. Registros		Núm. Bytes	Datos Byte 1	Datos Byte 2
10	04	01	00	01	02	00	32

Ejemplo 3: PDU de ModBus enviada para fijar Luminosidad al 50%

3.3.2. Valores a mostrar por el Marcador

Cada línea del Marcador tiene asociada una serie de registros, cada uno con una función distinta.

Hay que tener en cuenta que ciertas de éstas funciones no están disponibles en todos los Marcadores Alfanuméricos, en cuyo caso el valor escrito en los registros correspondientes será ignorado por el Marcador. Lo mismo sucede para los registros correspondientes a líneas no presentes en el Marcador.

Tener en cuenta que la numeración de líneas comienza con la 1 y en caso de haber más, la primera es la superior.

La relación de direcciones de los registros asociados a cada línea del marcador se detalla en la Tabla 11, donde para cada línea, los parámetros tienen las siguientes funciones:

- **Parámetro 1:** *Formato del Dato.* Indica cómo se va a interpretar los valores en los registros de datos (Valores 5 a 16). Las opciones se indican en la tabla siguiente.

Valor Parámetro 1	Formato	Valor Mínimo	Valor Máximo
0	Entero 16 bits con signo	-32768	+32767
1	Entero 16 bits sin signo	0	65535
2	Entero 32 bits con signo	-2147483647	+2147483647
3	Entero 32 bits sin signo	0	4294967295
16	Formato ASCII.	---	---

Tabla 7: Formato de los datos en función del Parámetro 1 de cada línea

- **Parámetro 2:** *Color del Texto de la línea.*

Valor Parámetro 2	Color del Texto
0	Color por defecto
1	VERDE
2	ROJO
3	AMBAR

Tabla 8: Color del texto en función del Parámetro 2 de cada línea

- **Parámetro 3:** *Modo de aparición del texto de la línea.*

Valor Parámetro 3	Modo de Aparición del Texto
0	Modo Normal o Inmediato
1	Intermitente
2	Intermitente Rápido

Tabla 9: Modo de aparición del texto en función del Parámetro 3 de cada línea

- **Parámetro 4:** *Posición del punto decimal. (Solo si Parámetro 1 tiene valor entre 0 y 3)*

Valor Parámetro 4	Posición del punto decimal
0	Sin Punto decimal
1	Punto decimal antes del digito 1: 0000000000.0
2	Punto decimal antes del digito 2: 000000000.00
3	Punto decimal antes del digito 3: 00000000.000
4	Punto decimal antes del digito 4: 0000000.0000
5	Punto decimal antes del digito 5: 000000.00000
6	Punto decimal antes del digito 6: 00000.000000
7	Punto decimal antes del digito 7: 0000.0000000
8	Punto decimal antes del digito 8: 000.00000000
9	Punto decimal antes del digito 9: 00.000000000
10	Punto decimal antes del digito 10: 0.0000000000

Tabla 10: Posición del punto decimal para cada valor del registro correspondiente a "Valor 3"

- **Parámetro 5 a 16:**

  Estos parámetros tienen distintas funciones, según el valor del Parámetro 1.

  - *Si Parámetro 1 tiene como valor 0 o 1 (Formato del Dato: Entero de 16 Bits):*
    - **Parámetro 5:** *Valor Numérico a representar*

  - *Si Parámetro 1 tiene como valor 2 o 3 (Formato del Dato: Entero de 32 Bits):*
    - **Parámetro 5:** *Valor Numérico a representar → Word (16 bits) Bajo*
    - **Parámetro 6:** *Valor Numérico a representar → Word (16 bits) Alto*

  - *Si Parámetro 1 tiene como valor 16 (Formato del Dato: ASCII):*
    - **Parámetros 5 a 16:** *Valor ASCII de los caracteres para representar. Cada Registro contiene 2 Bytes (cada uno de ellos con 1 carácter, siendo el LSB el primer carácter).*

      Se tienen en cuenta los caracteres hasta que nos encontramos con un carácter[6] NULL (0x00), que indica final de texto o llegamos al MSB del Parámetro 16.

      De forma que a través de MODBUS solo podremos enviar como máximo 24 caracteres (12 Registros x 2 Caracteres/Registro).

Línea	**Dirección Registro**								
	Par. 1	***Par. 2***	***Par. 3***	***Par. 4***	***Par. 5***	***Par. 6***	***Par. 7***	***Par. …***	***Par. 16***
1	410	411	412	413	414	415	416	…	41F
2	420	421	422	423	424	425	426	…	42F
3	430	431	432	433	434	435	436	…	43F
4	440	441	442	443	444	445	446	…	44F
5	450	451	452	453	454	455	456	…	45F
6	460	461	462	463	464	465	466	…	46F
7	470	471	472	473	474	475	476	…	47F
8	480	481	482	483	484	485	486	…	48F

[6] Recordad que en MODBUS, el número de Bytes de Datos debe ser par, de forma que hay que rellenar con caracteres NULL (0x00) los Bytes adicionales del final

9	490	491	492	493	494	495	496	…	49F
10	4A0	4A1	4A2	4A3	4A4	4A5	4A6	…	4AF
11	4B0	4B1	4B2	4B3	4B4	4B5	4B6	…	4BF
12	4C0	4C1	4C2	4C3	4C4	4C5	4C6	…	4CF
13	4D0	4D1	4D2	4D3	4D4	4D5	4D6	…	4DF
14	4E0	4E1	4E2	4E3	4E4	4E5	4E6	…	4EF
15	4F0	4F1	4F2	4F3	4F4	4F5	4F6	…	4FF

Tabla 11: Dirección de Registros asociados a cada línea del Marcador.
Valores en Hexadecimal

3.4. Ejemplos.

Para los siguientes casos, se muestra como sería la PDU de MODBUS enviada:

- Ejemplo 4: Mostrar en la línea 1 (la superior en caso de haber más de una) el valor 123.

Func.	Dirección Registro		Núm. Registros		Núm. Bytes	Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
10	04	10	00	05	0A	00	00	00	00	00	00	00	00	00	7B

Ejemplo 4: PDU de MODBUS enviada para mostrar '123' en la línea 1

Observando en detalle el campo de datos:

Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
00	00	00	00	00	00	00	00	00	7B
Formato 16 bits con signo		*Color por defecto*		*Sin Intermitencia*		*Sin Punto decimal*		*Valor 123*	

- Ejemplo 5: Mostrar en la línea 1 (la superior en caso de haber más de una) el valor 1.23 en color Verde.

Func.	Dirección Registro		Núm. Registros		Núm. Bytes	Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
10	04	10	00	05	0A	00	00	00	01	00	00	00	02	00	7B

Ejemplo 5: PDU de MODBUS enviada para mostrar '1.23'de color Verde en la línea 1

Observando en detalle el campo de datos:

Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
00	00	00	01	00	00	00	02	00	7B
Formato *16 bits con signo*		*Color Verde*		*Sin Intermitencia*		*Punto decimal* *antes del 2o dígito*		*Valor 123*	

- Ejemplo 6: Mostrar en la línea 1 (la superior en caso de haber más de una) el valor 1.23 en color Rojo y con Intermitencia.

Func.	Dirección Registro		Núm. Registros		Núm. Bytes	Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
10	04	10	00	05	0A	00	00	00	02	00	01	00	02	00	7B

Ejemplo 6: PDU de MODBUS *enviada para mostrar '1.23' en Rojo y con intermitencia en la línea 1*

Observando en detalle el campo de datos:

Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
00	00	00	02	00	01	00	02	00	7B
Formato *16 bits con signo*		*Color Rojo*		*Con Intermitencia*		*Punto decimal* *antes del 2o dígito*		*Valor 123*	

- Ejemplo 7: Mostrar en la línea 2 el valor 758 en color Rojo y con Intermitencia Rápida.

Func.	Dirección Registro		Núm. Registros		Núm. Bytes	Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10
10	04	20	00	05	0A	00	00	00	02	00	02	00	00	02	F6

Ejemplo 7: PDU de MODBUS *enviada para mostrar '758' en Rojo y con intermitencia Rápida en la línea 2*

Observando en detalle el campo de datos:

Datos	Datos	Datos	Datos	Datos	Datos	Datos	Datos	Datos	Datos

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10
00	00	00	02	00	02	00	00	02	F6
Formato *16 bits con signo*		*Color Rojo*		*Con Intermitencia Rápida*		*Sin Punto decimal*		*Valor 758*	

- Ejemplo 8: Mostrar en la línea 1 el texto "Hola".

Func.	Dirección Registro		Núm. Registros		Núm. Bytes	Data Byte 1	Data Byte 2	Data Byte 3	Data Byte 4	Data Byte 5	Data Byte 6	Data Byte 7	Data Byte 8	Data Byte 9	Data Byte 10	Data Byte 11	Data Byte 12	Data Byte 13	Data Byte 14
10	04	10	00	07	0E	00	10	00	00	00	00	00	00	48	6F	6C	61	00	00

Ejemplo 8: PDU de MODBUS *enviada para mostrar 'Hola' en la línea 1*

Observando en detalle el campo de datos:

Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	Datos Byte 10	Datos Byte 11	Datos Byte 12	Datos Byte 13	Datos Byte 14
00	10	00	00	00	00	00	00	48	6F	6C	61	00	00
Formato *ASCII*		*Color por defecto*		*Sin Intermitencia*		*NO USADO*		*'H'*	*'o'*	*'l'*	*'a'*	*NULL*	*NULL*

4. PROTOCOLO ASCII

4.1. Descripción del Protocolo ASCII.

Para poder comunicar mediante el Protocolo ASCII con un Marcador de MP Electronics, es necesario que éste disponga de un puerto de comunicación compatible con dicho protocolo y que el Protocolo se encuentre habilitado. Para habilitar / deshabilitar el Protocolo ASCII se debe acceder a la Configuración Interna del Marcador (Parámetro Núm. 230) y activar/desactivar el *Flag* correspondiente. La tabla 12 muestra los distintos puertos de comunicación de los Marcadores.

Puerto	***Protocolo ASCII***	
	Conectividad	***Habilitado por defecto***
RS232	SI	NO
RS485	SI	NO
USB	NO	---
ETHERNET (TCP)	NO	---

Tabla 12: Opciones de conectividad para el Protocolo ASCII

4.2. Opciones del Protocolo ASCII.

Con este protocolo el Marcador se comporta como Esclavo, es decir espera recibir una Trama y una vez recibida, si es conforme visualiza el valor. La trama es configurable para poder adaptarse a multitud de protocolos que utilizan el formato ASCII.

Para comprender como se puede configurar el protocolo se describen los términos utilizados:

- *Trama:* Está formada por todos los bytes necesarios para poder visualizar un valor. Para cada trama recibida correctamente, el marcador se actualizará. Cada trama consta de tres partes: La cabecera o header, los Datos y el final de trama.
- *Cabecera de la Trama o Header:* Se utiliza para identificar el comienzo de la trama. Se puede configurar entre 5 formatos distintos.
- *Campo de Datos:* Contiene la información que se debe visualizar. Mediante la configuración interna del Marcador, es posible seleccionar la parte del campo de datos que se desea visualizar.
- *Final de la Trama:* Se utiliza para identificar la llegada completa de la trama. Se puede configurar entre 6 tipos de final de trama.

A continuación, se detallan las opciones para cada parte de la trama. Los símbolos utilizados son los siguientes:

- **AH** → Carácter del digito de las Decenas de la Dirección del Marcador (‘0’ a ‘9’)[7].
- **AL** → Carácter del digito de las Unidades de la Dirección del Marcador (‘0’ a ‘9’).
- ***0x02*** → Valor 2 en hexadecimal
- ***0x03*** → Valor 3 en hexadecimal
- ‘@’ → Carácter '@' (código hexadecimal 0x40)
- ‘*’ → Carácter '*' (código hexadecimal 0x2A)
- CR → Carácter CR (código hexadecimal 0x0D)
- LF → Carácter LF (código hexadecimal 0x0A)
- ‘E’ → Carácter 'E' (código hexadecimal 0x45)
- ‘D’ → Carácter 'D' (código hexadecimal 0x44)
- ‘0’ → Carácter '0' (código hexadecimal 0x30)
- ACK → Carácter ACK (código hexadecimal 0x06)

4.2.1. Cabecera de la Trama.

La cabecera de la Trama son los Bytes iniciales que el Marcador espera recibir. Hay 5 opciones de Cabecera configurables:

- Cabecera Tipo 0: ‘@’ AH AL ‘E’ ‘D’
- Cabecera Tipo 1: *0x02*
- Cabecera Tipo 2: *0x02* AH AL
- Cabecera Tipo 3: *0x02* AL AH
- Sin Cabecera

En aquellas opciones en que se transmite la dirección del Marcador, ésta se conforma con los dígitos de decenas y unidades (en formato ASCII). Por lo tanto la dirección solo podrá comprender los valores (1 a 99). En caso de tener configurada una dirección del Marcador superior a 99, éste responderá a tramas dirigidas al dispositivo con dirección 99.

Por ejemplo, si la dirección del Marcador es 15, las cabeceras enviadas serán las que se muestran la siguiente tabla:

[7] La Dirección del Marcador debe estar en el rango 1 a 99. En caso de tener configurada una dirección mayor, el Marcador responderá a la dirección 99.

	Cabecera Tipo 0					Cabecera Tipo 1	Cabecera Tipo 2			Cabecera Tipo 3		
	'@'	**AH**	**AL**	**'E'**	**'D'**	***0x02***	***0x02***	**AH**	**AL**	***0x02***	**AL**	**AH**
Caracteres enviados	'@'	**'1'**	**'5'**	'E'	'D'	*0x02*	*0x02*	**'1'**	**'5'**	*0x02*	**'5'**	**'1'**
Caracteres enviados (ASCII)	40	**31**	**35**	45	44	02	02	**31**	**35**	02	**35**	**31**

Ejemplo 9: Diferentes cabeceras de trama enviadas al Marcador con dirección 15

4.2.2. Campo de Datos.

El Campo de datos contiene la información que se debe visualizar. Mediante la configuración interna del Marcador, es posible seleccionar la parte del campo de datos que se desea visualizar. Los parámetros configurables son los siguientes.

4.2.2.1. Saltar Caracteres.

Permite saltar caracteres del bloque de datos para poder seleccionar un valor dentro de una frase. Un ejemplo ayudara a entender el funcionamiento:

Supongamos que disponemos de un equipo que envía al Marcador la siguiente frase: "La Velocidad es de 147 km/h". Para poder visualizar el valor numérico de la velocidad debemos extraer del texto únicamente el valor 147. La forma de hacerlo es SALTANDO los caracteres del texto (Los espacios, los puntos y las comas cuentan como caracteres).

En este caso 19. Este es el valor que se debería configurar.

El rango de valores aceptado es desde 0 a 99.

4.2.2.2. Posición de los Valores.

Los dispositivos transmisores de la información a visualizar, pueden enviar primero el dígito de menor peso (unidades) mientras que otros envían primero el dígito de mayor peso. Este parámetro permite poder visualizar correctamente los valores adaptándose a todos los equipos.

- Posición Normal: El dígito que está en la posición seleccionada (1 a 9) de los datos se coloca en el dígito de la derecha del marcador. Los dígitos anteriores en la trama se van colocando a su izquierda.
- Posición Inversa: El dígito que está en la posición seleccionada (1 a 9) de los datos se coloca en el dígito de la derecha del marcador. Los siguientes dígitos de la trama se van colocando a su izquierda

Veamos un ejemplo.

Enviamos una Trama con el siguiente Campo de Datos. El Marcador, según éste parámetro visualizará unos valores u otros.

Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9
‘1’	‘2’	‘3’	‘4’	‘5’	‘6’	‘7’	‘8’	‘9’
31	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*

Ejemplo 10: Campo de datos de la Trama enviada

En caso de estar configurado el parámetro de la Posición de los Valores a Posición Normal y Dígito 9 seleccionado, el Marcador mostrará lo siguiente.

‘1’	**‘2’**	**‘3’**	**‘4’**	**‘5’**	**‘6’**	**‘7’**	**‘8’**	**‘9’**

Ejemplo 10: Valores Visualizados en Marcador de 9 Dígitos y Configuración Posición Normal (Dígito 9)

‘5’	**‘6’**	**‘7’**	**‘8’**	**‘9’**

Ejemplo 10: Valores Visualizados en Marcador de 5 Dígitos y Configuración Posición Normal (Dígito 9)

En caso de estar configurado el parámetro de la Posición de los Valores a Posición Inversa y Dígito 1 seleccionado, el Marcador mostrará lo siguiente.

‘9’	**‘8’**	**‘7’**	**‘6’**	**‘5’**	**‘4’**	**‘3’**	**‘2’**	**‘1’**

Ejemplo 10: Valores Visualizados en Marcador de 9 Dígitos y Configuración Posición Inversa (Dígito 1)

‘5’	**‘4’**	**‘3’**	**‘2’**	**‘1’**

Ejemplo 10: Valores Visualizados en Marcador de 5 Dígitos y Configuración Posición Inversa (Dígito 1)

4.2.3. Final de la Trama.

El Final de Trama son los Bytes finales que indican que la trama se ha terminado. Hay 6 opciones de Final de Trama configurables:

- Final de Trama Tipo 0: ‘*’ CR
- Final de Trama Tipo 1: *0x03*
- Final de Trama Tipo 2: CR LF

- Final de Trama Tipo 3: LF CR
- Final de Trama Tipo 4: CR
- Final de Trama Tipo 5: LF

4.2.4. Respuesta del Marcador.

Luego de recibir correctamente una Trama dirigida a él, el Marcador responderá con una trama de respuesta. Hay dos opciones de Respuesta del Marcador y la opción que éste no envíe ninguna respuesta.

- Sin Respuesta
- Respuesta del Marcador: Opción 1: '@' AH AL 'E' 'D' '0' '*' CR
- Respuesta del Marcador: Opción 2: *Cabecera de la Trama* ACK *Final de la Trama*

  Se envía una trama formada por la Cabecera de Trama configurada para la recepción, el carácter ACK (*0x06*) y el Final de Trama configurado para la recepción.

4.3. Ejemplos.

A continuación, se muestran ejemplos para determinadas configuraciones.

- Ejemplo 11:

  El Marcador con dirección 1[8], se configura de la siguiente forma:

Cabecera	***Saltar Caracteres***	***Posición Valores***	***Fin de Trama***	***Respuesta***
Sin Cabecera	0	Normal. Digito 9	Tipo 1 (*0x03*)	Sin Respuesta

Ejemplo 11: Configuración del Marcador

Al Marcador se le envía la siguiente trama:

Campo de Datos									**Final Trama**
Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	***0x03***
31	32	33	34	35	36	37	38	39	03

Ejemplo 11: Trama enviada al Marcador

[8] En este caso la Dirección del Marcador es irrelevante

El Marcador visualiza:

‘1’	‘2’	‘3’	‘4’	‘5’	‘6’	‘7’	‘8’	‘9’

Ejemplo 11: Valores Visualizados en Marcador de 9 Dígitos

- Ejemplo 12:

El Marcador con dirección 1, se configura de la siguiente forma:

Cabecera	***Saltar Caracteres***	***Posición Valores***	***Fin de Trama***	***Respuesta***
Tipo 2 (*0x02* AH AL)	0	Normal. Digito 9	Tipo 1 (*0x03*)	Tipo 3 (Cabecera ACK Fin de Trama)

Ejemplo 12: Configuración del Marcador

Al Marcador se le envía la siguiente trama:

Cabecera			**Campo de Datos**									**Final Trama**
0x02	**AH**	**AL**	Datos Byte 1	Datos Byte 2	Datos Byte 3	Datos Byte 4	Datos Byte 5	Datos Byte 6	Datos Byte 7	Datos Byte 8	Datos Byte 9	***0x03***
02	30	31	31	32	33	34	35	36	37	38	39	03

Ejemplo 12: Trama enviada al Marcador

El Marcador visualiza:

‘1’	‘2’	‘3’	‘4’	‘5’	‘6’	‘7’	‘8’	‘9’

Ejemplo 12: Valores Visualizados en Marcador de 9 Dígitos

El Marcador responde con la siguiente trama:

Cabecera			**Campo de Datos**	**Final Trama**
0x02	**AH**	**AL**	ACK	***0x03***
02	30	31	06	03

Ejemplo 12: Respuesta del Marcador

Anexo 1. Configuración por defecto de los Marcadores Alfanuméricos

Salvo que se indique lo contrario[9], los Marcadores Alfanuméricos de MP Electronics, en los parámetros que hacen referencia a las comunicaciones, tienen al salir de fábrica la siguiente configuración.

PARÁMETRO DE CONFIGURACIÓN	Valor por defecto
Dirección del Dispositivo	1.1
Protocolos de Comunicación	
Protocolo DTP	DESHABILITADO
Protocolo ELEKTRA	HABILITADO
Protocolo MODBUS	DESHABILITADO
Protocolo ASCII	DESHABILITADO
Puerto Serie 1	
Puerto Serie 1: Bauds	9600
Puerto Serie 1: Data Bits	8
Puerto Serie 1: Stop Bits	1
Puerto Serie 1: Paridad	Sin Paridad
Comunicación TCP / IP	
Dirección IP	192.168.1.100[10]
Máscara de Red	255.255.255.0
Puerta de Enlace	192.168.1.1
Servidor DNS Primario	192.168.1.100
Servidor DNS Secundario	192.168.1.100
Dirección IP por defecto	192.168.1.100
Máscara de Red por defecto	255.255.255.0
Direccionamiento IP Dinámico. Cliente DHCP Habilitado	SI
Protocolo ASCII	

[9] Consulte con la documentación especifica del producto MP adquirido

[10] Al estar el Cliente DHCP Activado, la Dirección IP del dispositivo será la que le asigne el Servidor DHCP de la Red Local a la que se conecte.

PARÁMETRO DE CONFIGURACIÓN	Valor por defecto
Cabecera de la Trama	SIN CABECERA
Final de Trama	Tipo 1 (*0x03*)
Respuesta	SIN RESPUESTA
Saltar Caracteres	0
Posición de los Valores	Normal. Digito 1

Tabla 13: Configuración por defecto de los Marcadores Numéricos MP Electronics